CLAIMS

1. Apparatus for processing data, said apparatus comprising:

a processor operable in a plurality modes and a plurality of domains, said plurality of domains comprising a secure domain or a non-secure domain, said plurality of modes including:

at least one secure mode being a mode in said secure domain; and

at least one non-secure mode being a mode in said non-secure domain;

wherein

when said processor is executing a program in a secure mode said program has access to secure data which is not accessible when said processor is operating in a non-secure mode;

said processor is responsive to one or more exception conditions for triggering exception processing using an exception handler, said processor being operable to select said exception handler from among a plurality of possible exception handlers in dependence upon whether said processor is operating in said secure domain or said non-secure domain.

2. Apparatus as claimed in claim 1, wherein at least one of said exceptions is a selectable exception handled by a selectable one of either a non-secure exception handler operating in a non-secure mode or a secure exception handler operating in a secure mode; and

at least one of said exceptions is a dedicated secure exception that is handled by a secure exception handler operating in a secure mode.

3. Apparatus as claimed in claim 1, wherein said one or more exception conditions can be programmably configured to trigger either a non-secure exception handler operating in a non-secure mode or a secure exception handler operating in a secure mode with any change of domain also being triggered when required.

4. Apparatus as claimed in claim 1, having a secure exception is triggered by one of a signal on a dedicated secure exception signal input and a non-secure exception signal input.

5. Apparatus as claimed in claim 1, having an exception signal input shared between secure and non-secure exceptions and a further input signal cooperating with said exception signal input to control whether a secure exception handler or a non-secure exception handler is triggered.

6. Apparatus as claimed in claim 1, wherein said secure exception handler is part of a secure operating system operable in said secure mode.

7. Apparatus as claimed in claim 1, wherein said non-secure exception handler is part of a non-secure operating system operable in said non-secure mode.

8. Apparatus as claimed in claim 1, wherein said processor is also operable in a monitor mode and any switching between a secure mode and a non-secure mode required for handling of an exception takes place via said monitor mode, said processor being operable at least partially in said monitor mode to execute a monitor program to manage switching between said secure mode and said non-secure mode.

9. Apparatus as claimed in claim 8, wherein said monitor program is operable to save and restore context data defining processor status when switching between a secure mode and a non-secure mode to handle an exception.

10. Apparatus as claimed in claim 8, wherein said processor includes a register bank and said monitor program is operable to flush at least a portion of said register bank shared between said secure mode and said non-secure mode when switching from said secure mode to said non-secure mode such that no secure data held within said register bank may pass from said secure mode to said non-secure mode other than as permitted by said monitor program.

11. Apparatus as claimed in claim 1, wherein said exception conditions includes one of more of:

a secure interrupt signal exception;

a mode switching software interrupt signal;

a reset exception;

an interrupt signal exception;

a software interrupt signal;

an undefined instruction exception;

a prefetch abort exception;

a data abort exception; and

a fast interrupt signal exception.

12. Apparatus as claimed in claim 1, wherein said processor is responsive to an exception condition to select an exception handler in dependence upon an exception vector value associated with said exception condition and stored within an active exception vector table for said exception condition; and

said active exception vector table is one of a plurality of exception vector tables.

13. Apparatus as claimed in claim 12, wherein said plurality of exception vector tables include a secure exception vector table selectable in said secure mode and a non-secure exception vector table selectable in said non-secure mode.

14. Apparatus as claimed in claim 12, wherein said processor is also operable in a monitor mode and any switching between a secure mode and a non-secure mode said plurality of exception vector is performed via said monitor mode.

15. Apparatus as claimed in claim 14, wherein said plurality of exception vector tables include a monitor mode exception vector table.

16. Apparatus as claimed in claim 15, wherein said processor is responsive to one or more parameters specifying which of said exceptions should be handled by said monitor mode exception vector table.

17. Apparatus as claimed in claim 13, wherein said secure vector table is said active vector table in said secure mode and said non-secure vector table is said active vector table in said non-secure mode unless said one or more parameters specify that said monitor mode vector table is said active vector table of said exception condition.

18. Apparatus as claimed in claim 16, wherein at least one of said parameters is stored in an exception trap mask.

19. Apparatus as claimed in claim 18, wherein said exception control register is writable when said processor is in said monitor mode and said exception trap mask register is non-writable when said processor is not in said non-secure domain.

20. Apparatus as claimed in claim 13, wherein said secure exception vector table is writable when said processor is in a secure mode and said secure exception vector table is non-writable when said processor is in a non-secure mode.

21. Apparatus as claimed in claim 13, wherein a secure exception handler that is part of a secure operating system is used said secure mode.

22. Apparatus as claimed in claims 13, wherein a non-secure exception handler that is part of a non-secure operating system is used said non-secure mode.

23. Apparatus as claimed in claim 12, comprising a plurality of vector table base address pointer registers each storing a respective base address value for a corresponding one of said plurality of exception vector tables.

24. A method of processing data, said method comprising the steps of:

executing a program with a processor operable in a plurality of modes and a plurality of domains, said plurality of domains comprising a secure domain or a non-secure domain, said plurality of modes including:

at least one secure mode being a mode in said secure domain; and

at least one non-secure mode being a mode in said non-secure domain;

wherein

when said processor is executing a program in a secure mode said program has access to secure data which is not accessible when said processor is operating in a non-secure mode; and

in response to one or more exception conditions, triggering exception processing using an exception handler; said processor being operable to select said exception handler from among a plurality of possible exception handlers in dependence upon whether said processor is operating is said secure domain of said non-secure domain.

25. A method as claimed in claim 24, wherein at least one of said exceptions is a selectable exception handled by a selectable one of either a non-secure exception handler operating in a non-secure mode or a secure exception handler operating in a secure mode; and

at least one of said exceptions is a dedicated secure exception that is handled by a secure exception handler operating in a secure mode.

26. A method as claimed in claim 24, wherein said one or more exception conditions can be programmably configured to trigger either a non-secure exception handler operating in a non-secure mode or a secure exception handler operating in a secure mode with any change of domain also being triggered when required.

27. A method as claimed in claim 24, having a secure exception signal input and a non-secure exception signal input.

28. A method as claimed in claim 24 having an exception signal input shared between secure and non-secure exceptions and a further input signal co-operating with said exception signal input to control whether a secure exception handler or a non-secure exception handler is triggered.

29. A method as claimed in claim 24, wherein said secure exception handler is part of a secure operating system operable in said secure mode.

30. A method as claimed in claim 24, wherein said non-secure exception handler is part of a non-secure operating system operable in said non-secure mode.

31. A method as claimed in claim 24, wherein said processor is also operable in a monitor mode and any switching between a secure mode and a non-secure mode required for handling of an exception takes place via said monitor mode, said processor being operable at least partially in said monitor mode to execute a monitor program to manage switching between said secure mode and said non-secure mode.

32. A method as claimed in claim 31, wherein said monitor program is operable to save and restore context data defining processor status when switching between a secure mode and a non-secure mode to handle an exception.

33. A method as claimed in claim 31, wherein said processor includes a register bank and said monitor program is operable to flush at least a portion of said register bank shared between said secure mode and said non-secure mode when switching from said secure mode to said non-secure mode such that no secure data held within said register bank may pass from said secure mode to said non-secure mode other than as permitted by said monitor program.

34. A method as claimed in claim 24, wherein said at least one exception conditions includes one of more of:

a secure interrupt signal exception;

a mode switching software interrupt signal;

a reset exception;

an interrupt signal exception;

a software interrupt signal;

an undefined instruction exception;

a prefetch abort exception;

a data abort exception; and

a fast interrupt signal exception.

35. A method as claimed in claim 24, wherein said processor is responsive to an exception condition to select an exception handler in dependence upon an exception vector value associated with said exception condition and stored within an active exception vector table for said exception condition; and

said active exception vector table is one of a plurality of exception vector tables.

36. A method as claimed in claim 35, wherein said plurality of exception vector tables include a secure exception vector table selectable in said secure mode and a non-secure exception vector table selectable in said non-secure mode.

37. A method as claimed in claim 35, wherein said processor is also operable in a monitor mode and any switching between a secure mode and a non-secure mode said plurality of exception vector is performed via said monitor mode.

38. A method as claimed in claim 37, wherein said plurality of exception vector tables include a monitor mode exception vector table.

39. A method as claimed in claim 37, wherein said processor is responsive to one or more parameters specifying which of said exceptions should be handled by said monitor mode exception vector table.

40. A method as claimed in claim 36, wherein said secure vector table is said active vector table in said secure mode and said non-secure vector table is said active vector table in said non-secure mode unless said one or more parameters specify that said monitor mode vector table is said active vector table of said exception condition.

41. A method as claimed in claim 39, wherein at least one of said parameters is stored in an exception trap mask register.

42. A method as claimed in claim 41, wherein said exception control register is writable when said processor is in said monitor mode and said exception trap mask register is non-writable when said processor is not in said monitor mode.

43. A method as claimed in claim 36, wherein said secure exception vector table is writable when said processor is in a secure mode and said secure exception vector table is non-writable when said processor is in a non-secure mode.

44. A method as claimed in claim 36, wherein a secure exception handler that is part of a secure operating system is used said secure mode.

45. A method as claimed in claims 36, wherein a non-secure exception handler that is part of a non-secure operating system is used said non-secure mode.

46. A method as claimed in claim 35, comprising storing within a plurality of vector table base address registers respective base address values for corresponding ones of said plurality of exception vector tables.

47. A computer program product having a computer program operable to control a data processing apparatus in accordance with the method of claim 24.